基于云环境下新产品开发团队优选研究 *

陈友玲，王 龙†，左丽丹，牛禹霏

(重庆大学 机械传动国家重点实验室，重庆 400044)

摘 要：针对云环境下新产品开发团队优选问题，充分考虑团队的研发能力、协调能力和服务质量，建立了包含知识相似度评价模型、协同效应评价模型和服务质量评价模型的综合评价模型；对算法的适应度函数和搜索方式加以改进，提出了改进的人工蜂群算法，对所建立的模型进行求解，为服务需求方选择出最优团队组合；最后，以自动引导运输车（AGV）的新产品开发为例，通过模型求解和算法对比，验证了所提方法的可行性和有效性。

关键词：团队选择；知识相似度；协同效应；服务质量；人工蜂群算法

中图分类号：TP391 doi: 10.19734/j.issn.1001-3695.2018.04.0442

Research on team optimal selection based on cloud environment for new product development

Chen Youling, Wang Long†, Zuo Lidan, Niu Yufei

(*The State Key Laboratory of mechanical Transmission, Chongqing University, Chongqing* 400044, *China*)

Abstract: For the new product development team optimal selection problems in cloud environment, the team's research and development capabilities, coordination capabilities, and quality of service were fully considered. An integrated evaluation model was established, which includes knowledge similarity evaluation model, collaborative effect evaluation model and quality of service evaluation model. The fitness function and search mechanism were improved and a modified artificial bee colony algorithm was proposed to solve the established model and find an optimal team combination for the service demanding party. Finally, taking a new product development of the automatic guided vehicle as an example, the feasibility and effectiveness of the proposed method are verified through the solution of model and the comparison of algorithm.

Key words: team optimal selection; knowledge similarity; collaborative effect; quality of service; artificial bee colony algorithm

0 引言

随着云计算和大数据技术的飞速发展，其应运而生的云平台也得到了广泛应用。它是利用互联网信息技术实现资源的虚拟化和共享化，充分利用资源，降低其因闲置而造成的浪费，从而达到“分散的资源集中使用，集中资源分散服务”的状态[1]。而人力资源作为云平台资源的重要组成部分，其跨区域实现能力的共享与协同，是协同完成云环境下新产品开发任务的基础。

以人力资源为对象，云平台通过跨地域、跨专业、跨行业的进行成员选择，组成临时团队，通过相互协作和知识共享完成设计任务[2]。但是对于复杂任务的设计和开发，仅仅通过成员选择已略显复杂，且无法满足任务对人力资源的需求，而以团队为选择单元进行团队组合是一个突破点。

云平台运营方拥有海量虚拟化的人力资源信息，其中包含大量服务相同或相近但服务质量和能力不同的服务团队[3]。如何快速地从海量组合方案中挑选出满足服务需求方的最佳团队组合，已成为云环境下团队组合优选的关键问题。现有的文献通过建立优选指标体系和选择优选方法对人力资源进行调用。陈友玲等人[4]基于成员知识相似度和学习能力，分阶段构建成员选择模型，并进行成员指派。姜艳萍等人[5]基于成员综合素质和专业技能指标期望，构建出交叉功能团队成员选择优化模型，完成团队成员选择。王娟茹等人[6]结合层次分析法和模糊优先规划法，求解虚拟团队成员优选问题。Zhang 等人[7]考虑成员的综合能力和人际关系，建立人际关系模型，采用多目标粒子群算法进行求解。刘敬等人[2]基于成员个人能力和协同合作能力信息，构建出优选指标体系和多目标优化模型，采用 SPEA2 算法进行求解。冯博等人[8]基于伙伴间协调关系和协同效应，建立团队伙伴选择的数学模型，通过 GRASP 启发式算法进行求解。Feng 等人[9]基于成员个人表现和成员间协作表现，建立团队成员优选模型，采用改进的非支配排序遗传算法进行

收稿日期：2018-04-30；**修回日期：**2018-06-11 **基金项目：**国家自然科学基金资助项目（71271224，71671020）

作者简介：陈友玲（1964-），女，重庆人，教授，博导，主要研究方向为基础工业工程、生产管理；王龙（1994-），男（通信作者），安徽合肥人，硕士研究生，主要研究方向为云制造、知识管理、新产品开发（wanglong7315@163.com）；左丽丹（1994-），女，硕士研究生，主要研究方向为管理科学、知识管理；牛禹霏（1993-），女，硕士研究生，主要研究方向为云制造，智慧制造.

求解。

以上文献主要从人力资源组合优选模型和求解方法进行了研究，并取得了大量的研究成果，在很大程度上解决了人力资源组合优选问题，促进了最优的人力资源组合方案选取。然而上述文献[4~7]是通过构建成员评价体系进行团队成员选取，但未考虑成员之间协同效应，可能造成实际选取结果不佳。文献[8,9]同时考虑团队成员信息以及成员间的协同关系进行虚拟团队组建，但在处理复杂任务时，单个成员选取已无法满足任务的人力资源需求。此外，上述文献采用了多目标粒子群优化算法、SPEA2 算法、GRASP 启发式算法和非支配排序遗传算法等，但在实际运行过程中，仍存在收敛速度不稳定、收敛效果差、易收敛于局部最优解等问题。

基于以上分析，本文以团队为选择对象，分析了云环境下团队组合特点，充分考虑团队的研发能力、协调能力和服务质量，建立了包含知识相似度评价模型、协同效应评价模型和服务质量评价模型的综合评价模型。通过改进人工蜂群算法，提高算法收敛速度和效果，为服务需求方求解出最优团队组合，完成服务需求方任务。

1　团队组合优选与知识转移过程描述

1.1　云平台团队组合与优选过程描述

云平台的团队选择过程如图 1 所示。在云平台上存在三个主体，可以划分为服务需求方、云平台运营方和资源提供方。针对云平台新产品开发各团队优选过程，其中服务需求方为新产品开发任务发布者，而资源提供方为各个设计研发团队，即服务需求方需要的为人力资源，而资源提供方供给的正是人力资源。从提交新产品开发任务请求至云平台到获得任务执行结果，要经历四个阶段，即任务分解阶段、子任务分析与搜索匹配阶段、团队组合优选阶段、执行和反馈阶段。在任务分解阶段，新产品开发任务 *Task* 提交到云平台，云平台根据服务需求方提供的相关信息和新产品开发的复杂性，将新产品开发任务分解为 N 个子开发任务 $(Subtask, ST)$，且每个 ST_i 能够被单个服务团队完成，$Task=\{ST_1,ST_2,\cdots,ST_i,\cdots ST_N\}$。在子任务分析与搜索匹配阶段，云平台分析每个 ST_i 的要求和特点，为其搜索匹配，得到满足 ST_i 的候选团队集合 $CTS_i=\{CT_i^1,CT_i^2,\cdots,CT_i^j,\cdots CT_i^M\}$。在团队组合优选阶段，需要从每个 ST_i 的候选团队集合 CTS_i 中选取一个团队来执行总开发任务 *Task*，可以得到团队优选组合 $TPCS=\{CT_1^{j_1},CT_2^{j_2},\cdots,CT_i^{j_i},\cdots CT_N^{j_N}\}$，其中 $CT_i^{j_i}\in CTS_i$，该团队组合也就是在团队组合优选阶段所要得到的团队组合执行路径。在执行和反馈阶段，由服务需求方确认后，将调用该团队组合来执行新产品开发任务，整个开发过程将由云平台进行监控和反馈。

本文主要注重团队组合优选阶段的有效解决方案。云平台通过多方考虑和全局优化，从每个候选团队集合中选择出一个团队，然后依次调用所选择的团队，构建合并出团队组合执行路径。

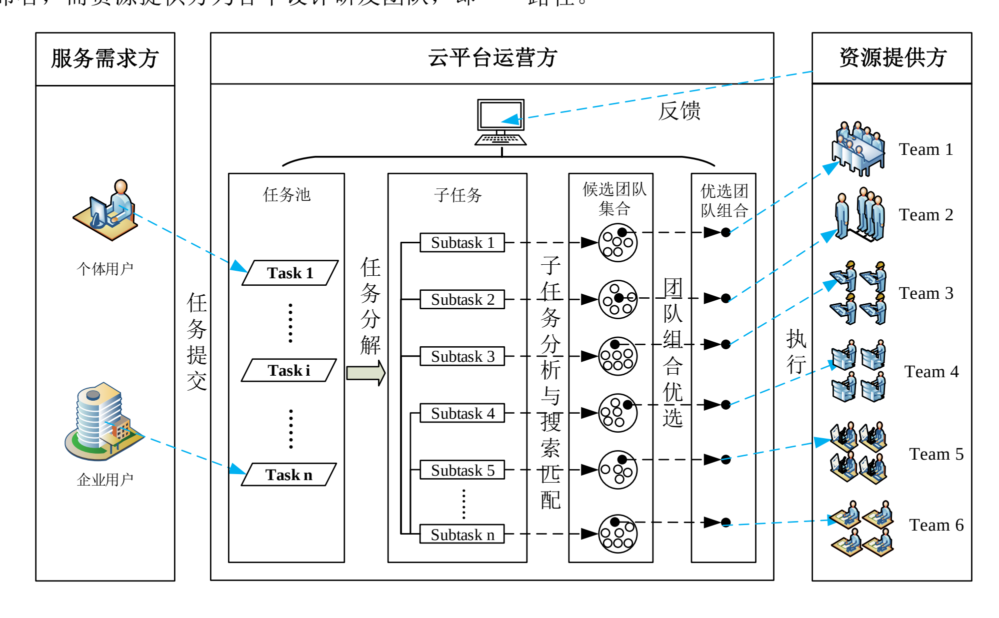


图 1　团队组合优选过程示意图

Fig.1　Team combination optimization process diagram

1.2　并行新产品开发与团队知识转移过程描述

通过相关文献查阅，对新产品开发过程中各子开发任务间的执行顺序进行研究。根据子任务执行的时间先后顺序，将开发过程中的设计活动分为串行模式、重叠模式和并行模式三种基本模式[10]。在串行模式中，任务 B 在任务 A 全部完成后再根据任务 A 传递的信息开始执行；在重叠模式中，任务 A 还未完成时，任务 B 根据任务 A 预发布信息，提前执行任务 B；在并行模式中，任务 A 和 B 同时开始执行，并进行信息交流传递。上述三种模式在完成任务的周期和成本各不相同，串行模式的产品开发周期长，信息交流次数少，开发成本低，返工可能性

高；重叠模式的产品开发周期适中，存在关联任务间的信息交流、开发成本和返工可能性中等；并行模式的产品开发时间短，信息交流次数多，开发成本高，返工概率随交流次数增多而减少。这几种开发模式间接地反映了开发时间和开发成本间的权衡，而本文研究对象为并行模式的新产品开发。新产品开发模式如图 2 所示。

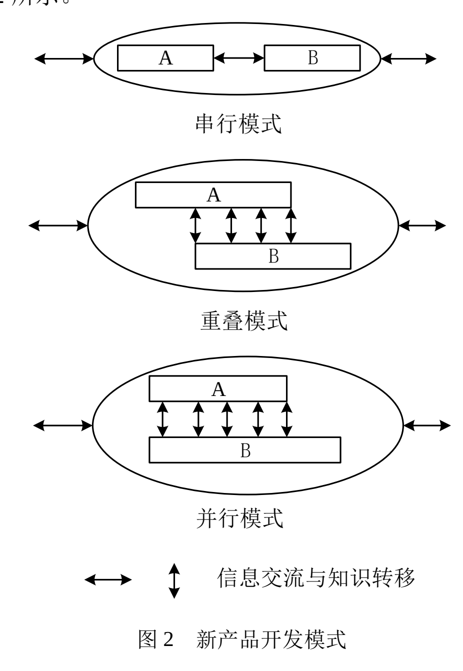


图 2　新产品开发模式

Fig.2　New product development model

新产品开发过程实质上是一个知识创造、知识积累和知识转移的过程[11]，而产品开发团队正是知识的提供者，知识转移过程是通过团队间的信息交流完成的。以两个开发任务并行开发为例，知识转移过程如图 3 所示。团队 a 和 b 分别完成任务 A 与 B，在任务开发周期内，团队 a、b 间共进行了 n 次知识转移。在团队 a 与 b 进行第一次知识转移之前，由其原始积累知识进行知识作业；在第一次知识转移后，团队 a 和 b 根据获取的初始知识进行知识作业；在第二次知识转移后，由于在上一次知识转移过程中获取错误知识或者不完备知识，需在本次知识转移过程中更新错误知识或完善知识，从而进行知识返工[12]，另外团队 a、b 会根据更新的知识进行新的知识作业；之后进行第三次知识转移，以此类推，直至完成任务开发。子开发任务的时间主要由知识作业时间、知识返工时间和交流时间构成。知识作业时间为知识转移前后知识作业所耗费的时间，它主要由任务本身决定，不受外界因素影响，用 ${}^{e}T_i$ 表示子任务 i 的总共的知识作业时间；知识返工时间为知识返工所消耗的时间，用 ${}^{r}T_{ij}^{k}$ 表示在子任务 i 与 j 的负责团队间第 k ($k\neq 1$)次知识转移中子任务 j 所产生的知识返工时间；沟通时间为两团队间知识转移所耗费的时间，用 ${}^{c}T_{ij}^{k}$ 表示子任务 i 与 j 的负责团队间第 k 次知识转移中的交流时间。此外，产品开发团队的研发能力、学习能力、沟通交流能力和协同能力等均是影响知识转移次数和作业时间、返工时间的主要因素。

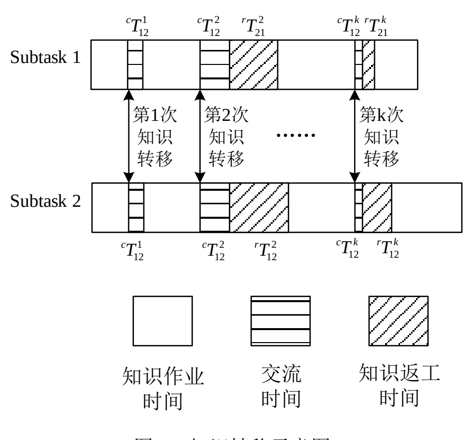


图 3　知识转移示意图

Fig.3　Schema diagram of knowledge transfer

2　模型构建

为了全面评价团队优选组合，本文从研发能力、协同效应和服务质量三个方面分别构建出知识相似度评价模型、协同效应评价模型和服务质量评价模型，并综合三个模型的优势进行有效结合，构建出综合评价模型，对团队优选组合进行全面、准确的评价，为服务需求方选择合适的团队组合提供依据。模型结构示意图如图 4 所示。

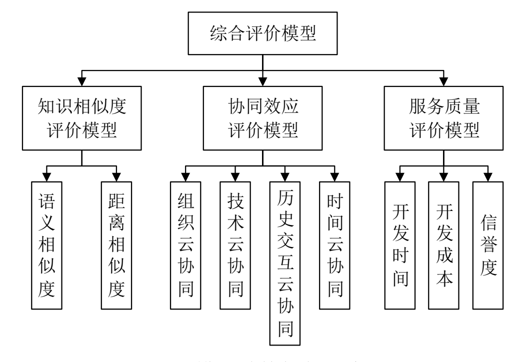


图 4　模型结构框架示意图

Fig.4　Schematic diagram of model structure

2.1　知识相似度评价模型

为了更好地衡量开发团队对于新产品开发任务的研发能力，本文提出知识相似度(knowledge similarity, KS)，指开发团队知识结构与开发任务知识需求之间的相似程度。知识相似度越大，表明开发团队越适合该开发任务，对此任务的开发能力越强，反之亦然。

本文参考文献[13]相关概念对知识相似度进行计算，给出以下相关概念。知识点(knowledge point, KP)，是新产品开发子任务中最小的数据单元，且无法再进一步划分。开发任务的知识集合中包含多个相关知识点，并根据相关知识点绘制知识结构树。知识模块(knowledge module, KM)，是上述知识点组成的集合。新产品开发任务的知识需求模块可表示为 $KMR=\{(kpr_1,\alpha_1),(kpr_2,\alpha_2),\cdots,(kpr_i,\alpha_i)\}$，其中：$kpr_i$ 为新产品开发任务所需要的知识点；α_i 为各知识点的需求程度。各候选团队知识掌握模块可表示为

$KMG=\{(kpg_1,\beta_1),(kpg_2,\beta_2),\cdots,(kpg_j,\beta_j)\}$，其中：$kpg_j$ 为开发团队掌握的相关知识点；β_i 表示团队对各知识点的掌握程度。根据新产品开发任务的知识需求模块 KMR 和候选团队知识掌握模块 KMG，定义新产品开发任务的知识需求向量空间为 $V_r=(\alpha_1,\alpha_2,\cdots,\alpha_i)$，候选团队知识掌握程度向量空间 $V_g=(\beta_1,\beta_2,\cdots,\beta_j)$。

知识相似度包括语义相似度和距离相似度[4]，其中语义相似度根据 KMR 和 KMG 知识模块中相关知识点在知识结构树中的位置决定，其计算公式如(1)所示；距离相似度由 V_r 和 V_g 两个向量空间的欧几里德距离大小决定，计算公式如(2)所示。为了更好的描述知识相似度，将语义相似度和距离相似度归约到区间(0,1]，其结果如式(3)(4)所示。

$$S_s(kpr_i,kpg_j)=\frac{\lambda\times|C_{sub}(kpr_i,R)\cap C_{sub}(kpg_j,R)|}{|C_{sub}(kpr_i,R)\cup C_{sub}(kpg_j,R)|}+\frac{(1-\lambda)\times|C_{super}(kpr_i,R)\cap C_{super}(kpg_j,R)|}{|C_{super}(kpr_i,R)\cup C_{super}(kpg_j,R)|} \tag{1}$$

$$d(V_r,V_g)=\begin{cases}\sqrt{\sum_{i=1}^{k}(\alpha_i-\beta_i)^2}, & \alpha_i>\beta_i;\\ 0, & \alpha_i\le\beta_i。\end{cases} \tag{2}$$

其中：λ 为知识结构树中父类与子类的相对程度；$|C_{sub}(kpr_i,R)\cap C_{sub}(kpg_j,R)|$ 为知识结构树下同属于 kpr_i 和 kpg_j 的子类层级数；$|C_{sub}(kpr_i,R)\cup C_{sub}(kpg_j,R)|$ 为树下下同属于 kpr_i 或 kpg_j 的子类层级数；$|C_{super}(kpr_i,R)\cap C_{super}(kpg_j,R)|$ 为树下包括 kpr_i 和 kpg_j 父类层级数；$|C_{super}(kpr_i,R)\cup C_{super}(kpg_j,R)|$ 树下包括 kpr_i 或 kpg_j 父类层级数；k 表示 $\alpha_i>\beta_i$ 的元素个数；当 $a\ne b$ 时，α_i 或 β_i 用 0 表示。

$$S_s=\frac{\sum_{i=1}^{a}\sum_{j=1}^{b}S_s(kpr_i,kpg_j)}{1+\sum_{i=1}^{a}\sum_{j=1}^{b}S_s(kpr_i,kpg_j)} \tag{3}$$

$$S_d=\frac{1}{1+d(V_r,V_g)} \tag{4}$$

由上述语义相似度式(3)和距离相似度式(4)可获得知识相似度，如式(5)所示。其中 γ 为语义相似度和距离相似度的相对权重。

$$S_k=\gamma\times S_s+(1-\gamma)\times S_d \tag{5}$$

通过计算新开发产品任务与服务团队间知识相似度，衡量团队与开发任务间的匹配程度。而团队优选组合的知识相似度如式(6)所示。其中 $S_{k,i}$ 表示候选团队 CT_i 与第 i 子开发任务的知识相似度。

$$S(TPCS)=\sum_{i=1}^{n}S_{k,i} \tag{6}$$

2.2　协同效应评价模型

团队知识相似度衡量团队的研发能力，而团队的协同效应则是衡量交互协调能力。团队协同效应通常指多个团队协调合作的效益大于其单个团队效益的总和[14]。在并行新产品开发团队组合中，只注重团队在处理各自任务的能力，往往忽略团队间的内在联系，导致开发的新产品无法达到预期要求。因此，本文将团队协同效应应用到并行新产品开发的团队选择中，追求团队组合的整体协同效应最佳。

本文对协同效应影响因素进行分析，结合并行情况下新产品开发团队组合的特性，归纳总结出团队组合协同效应的评价指标，如表 1 所示。

表 1　团队组合协同效应评价指标

Table 1　Team combination synergy evaluation index

评价指标	定义
组织云协同	组合内团队间工作流程和组织结构的协同程度
技术云协同	组合内团队间相关技术的协同程度
历史交互云协同	组合内团队间历史交互记录中的协同程度
时间云协同	组合内团队间响应时间和交流时间的协同程度

1)组织云协同(D1)　服务组合中各团队按照各自的工作流程和组织结构进行产品开发作业，流程同步性和组织一致性能够保证各阶段团队间的信息沟通和问题的及时调整。

2)技术云协同(D2)　在产品过程中需要各种专业技术和相关软件的支持，如果团队使用的相关软件不一致或者版本有差异，甚至专业技术不统一，会极大地阻碍团队间的信息传递和数据交换。

3)历史交互云协同(D3)　如果团队间拥有过合作的经历，每次合作完成后会对合作团队进行相应的评价，尤其是在云环境下，云平台会记录下历史交互的数据和信息。若双方有过合作历史记录，则根据历史交互数据进行协同效应评价；若无历史交互，则该项协同效应为 0。

4)时间云协同(D4)　团队的响应时间以及团队间交流时间节点是否一致，决定了团队间的契合度以及相关信息数据的及时更新和问题的及时发现，后续相关开发工作也会受到影响。

云平台储存了大量多方面的用户数据，并且具备及时更新和数据分析的能力，因此云平台专家可以利用大数据和数据挖掘技术结合云平台用户的历史交互数据进行协同效应分析，从而对上述的各协同效应评价指标值作出准确评估。

设上表中各评价指标的权重向量为 $W_c=(w_1,w_2,\cdots,w_4)^T$，且满足 $\sum_{i=1}^{4}w_i=1$，$0\le w_i\le1$。通过加权法则，可计算出团队组

合中各团队之间的协同效应值 $C_{CT_i,CT_j}=\sum_{i=1}^{4}w_i\times D_i$。其中：$C_{CT_i,CT_j}$ 为团队组合中候选团队 CT_i 与子团队 CT_j 之间的协同效应值；D_i 表示第 i 个协同效应评价指标。式(7)表示候选团队 CT_i 与组合内候选团队的平均协同效应值；式(8)表示团队组合的协同效应值。

$$C_{CT_i}=\frac{\sum_{j=1}^{n}C_{CT_i,CT_j}}{n-1},(i\neq j) \tag{7}$$

$$C(TPCS)=\left(\sum_{i=1}^{n}C_{CT_i}\right)\Big/n \tag{8}$$

2.3 服务质量评价模型

在团队组合中，需考虑整体组合的服务质量，即团队组合在完成并行新产品开发任务所预估花费的时间、成本以及团队自身的信誉度等指标。

根据 1.2 节中，两团队间的知识转移会产生知识返工时间、知识交流时间以及其相应产生的成本。针对并行新产品开发任务，根据文献[11]中所建立的模型，规定各团队的学习能力指数、知识发出方的发布错误知识概率和各子开发任务的开发难度以及任务间相对依赖程度，在追求全局收益最大化原则下建立知识转移时间数学模型，并求解团队间交流次数。根据知识返工函数求解出每次交流后的知识返工时间，并对知识交流和知识返工的成本进行计算。

开发时间 T 为各团队开始进行各自任务设计至完成总开发任务所耗费的时间。单个团队所花费的开发时间 T_i 包括知识作业时间 ${}^{e}T_i$、知识返工时间 ${}^{r}T_i$，交流时间 ${}^{c}T_i$。$T_i=T_{modified}\times\left({}^{e}T_i+{}^{r}T_i+{}^{c}T_i\right)$。其中时间修正系数 $T_{modified}$ 表示实际时间与预期时间之间的偏离程度，用于对开发时间计算的修正。$T_{modified}=\prod_{i=1}^{n}\left(1+\left(T_{actual}-T_{expected}\right)/T_{expected}\right)$，其中：$T_{actual}$ 指实际完成任务的时间；$T_{expected}$ 指预计完成任务的时间；n 指该团队完成服务需求方提交任务的总个数。开发成本 C 为各团队开始进行各自任务设计至完成总开发任务所产生的费用。单个团队的开发成本 C_i 包括知识作业成本 ${}^{e}C_i$、知识返工成本 ${}^{r}C_i$ 和交流成本 ${}^{c}C_i$，$C_i={}^{e}C_i+{}^{r}C_i+{}^{c}C_i$。信誉度 M 是指云平台对于服务团队的一种评价指标，反映团队的一种信誉程度。信誉度越高，服务需求方的合作意愿越强烈。云平台根据服务团队的历史匹配记录、履约状况等给出的一个综合性指标，且 $M\in[0,1]$。以上评价指标的选取、数据的采集和量化等方面已经进行了相关深入的研究，本文限于篇幅，不再赘述。

本文将服务质量信息组定义为 $Q=\{T,C,M\}$，在并行新产品开发团队组合的指标计算公式如表 2 所示。

表 2 团队组合指标计算公式

Table 2 Team combination indicator calculation formula

模式	参数		
	T	C	M
并行	$Max(T_i)$	$\sum_{i=1}^{n}C_i$	$\left(\sum_{i=1}^{n}M_i\right)\Big/n$

本文根据云平台专家组对服务质量的各个指标进行权重赋值，$W_q=(w_1,w_2,w_3)^T$，且满足 $\sum_{i=1}^{3}w_i=1$，$0\le w_i\le 1$。式(9)为团队优选组合的服务质量评价值，其中 T_N,C_N,M_N 分别表示团队优选组合时间指标、成本指标和信誉度指标归一化结果。

$$Q(TPCS)=w_1\times T_N+w_2\times C_N+w_3\times M_N \tag{9}$$

2.4 综合评价模型

团队优选组合的评价模型可以定义为一个三元组合 $G=\{S(TPCS),C(TPCS),Q(TPCS)\}$，其中：$S(TPCS)$ 表示团队组合的知识相似度；$C(TPCS)$ 表示团队组合的协同效应值，$C(TPCS)$ 表示团队组合服务质量评价值；$TPCS$ 表示团队优选组合。本文从以上三个方面去全面衡量团队组合的综合开发实力，充分考虑三者之间的关系，通过相关专家赋值，通过层次分析法计算，得到权重 $W_I=(w_1,w_2,w_3)^T$，且满足 $\sum_{i=1}^{3}w_i=1$，$0\le w_i\le 1$。构建出综合评价模型，根据式(10)计算出团队组合的综合评价值。

$$I(TPCS)=w_1\times S(TPCS)+w_2\times C(TPCS)+w_3\times Q(TPCS) \tag{10}$$

3 改进人工蜂群算法

人工蜂群（artificial bee colony，ABC）算法自提出之后，相关专家和研究人员在原有算法基础上加以改进，提高了算法的求解速度、稳定性和模型求解的适用性。而人工蜂群算法改进主要体现在适应度函数计算和蜜源搜索机制上。本文结合云环境下团队组合优选问题特性和所建立的求解模型特点，对 ABC 算法的适应度函数计算方式和蜜源位置更新方式进行改进，提出一种改进的人工蜂群（modified artificial bee colony，MABC）算法。

标准 ABC 算法中，蜜蜂拥有雇佣蜂、侦查蜂和跟随蜂三种角色。由雇佣蜂确定蜜源 X_i，并对其进行访问；跟随蜂根据雇佣蜂分享的蜜源信息，以概率值 P_i 选择跟随的雇佣蜂；在蜜源被抛弃后，由侦查蜂发现新蜜源[15]。以上为 ABC 算法中三种角色的简单介绍，下面将对标准 ABC 算法进行改进。

3.1 适应度函数

适应度函数引导群体进化的方向，影响算法迭代的次数和解的质量，不同的适应度函数会得出不同优劣程度的解。标准 ABC 算法使用文献[15]中复杂的适应度值计算函数，本文结合综合评价模型特点，直接用综合评价模型的目标函数的值来代替适应度值。因为综合评价模型中三个维度中知识相似度、协

同效应值和服务质量评价值的取值均在[0,1]内，且最终的综合评价值也类同于适应度值，能够判断蜜源的优劣，方便概率值 P_i 的计算。适应度函数计算方式如式(11)所示。

$$fit_i = f_i = W_I \bullet X_i \tag{11}$$

$$P_i = fit_i \Big/ \sum_{i=1}^{SN} fit_i \tag{12}$$

其中：fit_i 表示适应度函数；W_I 为综合评价模型中的权重值；X_i 表示蜜源 i，为问题的一个可行解，也就是团队组合产生的三维度量指标值；f_i 为目标函数在 X_i 上所取得的函数值；SN 为蜜源数量，P_i 为跟随蜂选择雇佣蜂的概率。

3.2　搜索方式

雇佣蜂和侦查蜂的搜索方式决定着蜜蜂找到蜜源的速度和准确性。标准 ABC 算法通过在蜜源附近随机选取另外的蜜源，体现了算法的随机性，但算法的求解精度和收敛速度收到了限制。为提高算法的各项性能，本文改进了新蜜源位置的搜索方式[16]，如式(13)所示。

$$v_{ij} = x_{ij} + c_1(t)\varphi_{ij}\left(x_{ij} - x_{kj}\right) + c_2(t)\phi_{ij}\left(x_{bj} - x_{ij}\right) \tag{13}$$

其中：v_{ij} 为代替 x_{ij} 的新蜜源；φ_{ij} 为[-1,1]间的随机数；ϕ_{ij} 为[0,1]间的随机数；t 为当前迭代次数；k 为[1,SN]间的随机整数；x_{bj} 为当前最优蜜源 X_b 的第 j 个参数；$c_1(t)$ 和 $c_2(t)$ 为两个自适应控制参数，用于平衡算法不同阶段的搜索。在算法早期，$c_1(t)$ 应为一个较大值，$c_2(t)$ 应为一个较小的值；随着迭代次数增加，$c_1(t)$ 逐渐减小，$c_2(t)$ 逐渐增大。在算法运行过程中，通过 $c_1(t)$ 和 $c_2(t)$ 的相互协作，实现算法运行过程的自我调节，并且由 $c_1(t)$ 和 $c_2(t)$ 的变化来调整搜索过程，克服人工蜂群算法易陷入局部最优等缺陷[17]。$c_1(t)$ 和 $c_2(t)$ 的计算如式(14)所示。

$$c_1(t) = c_1^{max}\left(\frac{c_1^{min}}{c_1^{max}}\right)^{\frac{t}{MCN}},\quad c_2(t) = c_2^{max}\left(\frac{c_2^{max}}{c_2^{min}}\right)^{\frac{t}{MCN}} \tag{14}$$

其中：t 为当前迭代次数；MCN 为最大迭代次数；c_i^{min} 和 c_i^{max} 为函数 $c_i(t)$ 的最小和最大值。

此外，根据上述所建立的综合评价模型，分析了团队组合优选问题特点，发现所求解模型的搜索空间具有较小的可行区域，很难随机产生可行解和靠近可行区域的非可行解。在一个蜜源被访问次数超过阈值 $limit$，侦查蜂会随机寻找新蜜源。为了改善这种随机情况，本文在改进其更新方式，如式(15)所示。

$$x_{ij} = x_{ij} + \varphi_{ij}\left(x_{kj} - x_{ij}\right) + \left(1 - \varphi_{ij}\right)\left(x_{bj} - x_{ij}\right) \tag{15}$$

其中：φ_{ij} 为[-1,1]间的随机数；k 为[1,SN]间的随机整数；x_{bj} 为当前最优蜜源 X_b 的第 j 个参数。

3.3　算法流程

根据以上改进之处对算法进行修改，基于改进人工蜂群算法的云环境下新产品开发团队组合优选流程如图 5 所示。

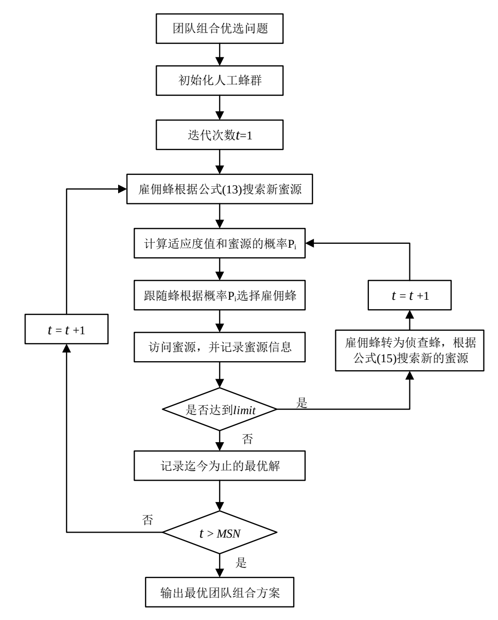


图 5　团队组合优选算法流程

Fig.5　Team combination optimization algorithm flow

4　实验结果与分析

为了验证 MABC 算法和综合评价模型在解决云环境下新产品开发团队组合优选问题上的可行性和有效性，本文通过有效数据采集和相关数据模拟，构建出实验算例。实验环境为 MATLAB R2015a，Windows10，2.60 GHz，8 GB RAM。

4.1　实验背景及相关数据求解

实验以 AGV 小车设计为背景，服务需求方向云平台提交 AGV 小车的新产品开发设计任务需求，云平台根据新产品开发任务要求和条件等相关信息，将设计任务分解成如图 6 所示的八个子开发任务，其子任务间的相互依赖程度如表 3 所示。同时，云平台进行搜索匹配，为每个子开发任务找到相关候选开发团队集合，其相关信息如表 4 所示。表中包括知识相似度模型处理后的各团队对子任务的知识相似度，知识相似度评价模型中取 γ=0.5。通过算法和模型进行求解，为服务需求方需找到最优团队组合，并由这些团队负责 AGV 小车的开发工作。

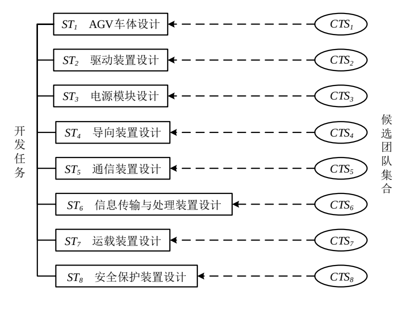


图 6　子开发任务示意图

Fig.6　Sub-development task diagram

表 3　子任务间相互依赖程度

Table 3　Degree of interdependence between subtasks

依赖程度	ST_1	ST_2	ST_3	ST_4	ST_5	ST_6	ST_7	ST_8
ST_1	—	0.35	0.28	0.15	0.12	0.20	0.55	0.68
ST_2	0.22	—	0.56	0.18	0.10	0.12	0.64	0.16
ST_3	0.24	0.48	—	0.14	0.15	0.20	0.44	0.08
ST_4	0.18	0.30	0.20	—	0.28	0.35	0.25	0.15
ST_5	0.10	0.22	0.25	0.16	—	0.58	0.10	0.05
ST_6	0.10	0.15	0.24	0.12	0.62	—	0.28	0.08
ST_7	0.35	0.52	0.32	0.15	0.05	0.06	—	0.18
ST_8	0.44	0.15	0.10	0.10	0.12	0.10	0.32	—

表 4　候选团队集相关参数

Table 4　Candidate team set related parameters

候选团队		知识相似度 S_k	开发时间 T		知识作业成本 ${}^{e}C$	信誉度 M
			${}^{e}T$	$T_{modified}$		
CTS_1	CT_1^1	0.82	282	1.04	675	0.88
	CT_1^2	0.77	294	1.01	690	0.86
CTS_2	CT_2^1	0.85	409	0.98	815	0.89
	CT_2^2	0.72	386	1.11	800	0.92
	CT_2^3	0.83	415	0.96	790	0.91
CTS_3	CT_3^1	0.84	400	0.95	975	0.84
	CT_3^2	0.89	410	0.92	950	0.92
	CT_3^3	0.90	423	0.93	980	0.93
CTS_4	CT_4^1	0.76	260	1.06	720	0.93
	CT_4^2	0.74	273	1.08	740	0.90
CTS_5	CT_5^1	0.79	308	0.99	830	0.88
	CT_5^2	0.78	320	1.02	800	0.91
CTS_6	CT_6^1	0.80	296	1.04	895	0.87
	CT_6^2	0.82	304	0.98	870	0.89
CTS_7	CT_7^1	0.85	251	0.95	555	0.92
	CT_7^2	0.88	230	1.08	530	0.94
CTS_8	CT_8^1	0.79	225	1.12	595	0.95
	CT_8^2	0.77	256	1.07	610	0.96

为了方便 MABC 算法对模型进行求解，对协同效应评价模型相关数据进行计算和预处理。在协同效应评价模型中，取权重向量 $W_c=(0.25,0.25,0.25,0.25)^T$，得到团队间协同效应值矩阵表，如表 5 所示。其中同一团队集合中各团队不进行协同效应评估，用“—”表示。此外，也需对服务质量评价模型中相关数据进行处理。利用文献[11]中所建立模型，根据表 3 和 4 相关信息，求解出在完成并行新产品开发任务后各团队间所进行团队交流和知识作业所耗费的时间和成本，其相关信息如表 6 所示，为团队组合优选提供数据支持。表中“3/7/12/14”分别表示“交流时间 ${}^{c}T$ /交流成本 ${}^{c}C$ /知识返工时间 ${}^{r}T$ /知识返工成本 ${}^{r}C$ ”。

表 5　团队间协同效应值矩阵表

Table 5　Inter-team synergy value matrix

	CT_1^1	CT_1^2	CT_2^1	CT_2^2	CT_2^3	CT_3^1	CT_3^2	CT_3^3	CT_4^1	CT_4^2	CT_5^1	CT_5^2	CT_6^1	CT_6^2	CT_7^1	CT_7^2	CT_8^1	CT_8^2
CT_1^1	—	—	0.43	0.52	0.64	0.51	0.73	0.41	0.58	0.53	0.57	0.68	0.47	0.57	0.56	0.63	0.65	0.48
CT_1^2	—	—	0.55	0.49	0.52	0.64	0.59	0.62	0.52	0.55	0.47	0.59	0.62	0.66	0.58	0.60	0.59	0.55

CT_2^1	0.43	0.55	—	—	—	0.60	0.59	0.62	0.61	0.49	0.59	0.65	0.52	0.49	0.57	0.62	0.66	0.62
CT_2^2	0.52	0.49	—	—	—	0.58	0.66	0.62	0.52	0.58	0.63	0.62	0.56	0.60	0.52	0.69	0.55	0.59
CT_2^3	0.64	0.52	—	—	—	0.64	0.69	0.60	0.59	0.55	0.61	0.66	0.55	0.58	0.63	0.71	0.70	0.60
CT_3^1	0.51	0.64	0.60	0.58	0.64	—	—	—	0.62	0.67	0.66	0.61	0.60	0.59	0.65	0.67	0.62	0.64
CT_3^2	0.73	0.59	0.59	0.66	0.69	—	—	—	0.70	0.66	0.65	0.68	0.58	0.62	0.67	0.69	0.63	0.59
CT_3^3	0.66	0.62	0.62	0.62	0.60	—	—	—	0.68	0.65	0.61	0.66	0.62	0.65	0.68	0.71	0.64	0.62
CT_4^1	0.58	0.52	0.61	0.52	0.59	0.62	0.70	0.68	—	—	0.65	0.68	0.63	0.69	0.58	0.64	0.68	0.60
CT_4^2	0.53	0.55	0.49	0.58	0.55	0.67	0.66	0.65	—	—	0.59	0.67	0.66	0.65	0.61	0.63	0.66	0.63
CT_5^1	0.57	0.47	0.59	0.63	0.61	0.66	0.65	0.61	0.65	0.59	—	—	0.60	0.62	0.52	0.53	0.62	0.60
CT_5^2	0.68	0.59	0.65	0.62	0.66	0.61	0.68	0.66	0.68	0.67	—	—	0.58	0.65	0.50	0.56	0.65	0.57
CT_6^1	0.47	0.62	0.52	0.56	0.55	0.60	0.58	0.62	0.63	0.66	0.60	0.58	—	—	0.67	0.68	0.69	0.72
CT_6^2	0.57	0.66	0.49	0.60	0.58	0.59	0.62	0.65	0.69	0.65	0.62	0.65	—	—	0.66	0.71	0.70	0.71
CT_7^1	0.56	0.58	0.57	0.52	0.63	0.65	0.67	0.68	0.58	061	0.52	0.50	0.67	0.66	—	—	0.58	0.49
CT_7^2	0.63	0.60	0.62	0.69	0.71	0.67	0.69	0.71	0.64	0.63	0.53	0.56	0.68	0.71	—	—	0.62	0.57
CT_8^1	0.65	0.59	0.66	0.55	0.70	0.62	0.63	0.64	0.68	0.66	0.62	0.65	0.69	0.70	0.58	0.62	—	—
CT_8^2	0.48	0.55	0.62	0.59	0.60	0.64	0.59	0.62	0.60	0.63	0.60	0.57	0.72	0.71	0.49	0.57	—	—

4.2 算法求解

根据上述相关数据初始化结果，通过 MABC 算法对团队组合优选的组合评价评价模型进行求解，并在相同参数设置和环境下重复进行 100 次实验。MABC 算法的相关数据设置如下：SN=100， MCN=500， $limit$=100， $c_1^{min}=0.8$， c_1^{max}=1.2， c_2^{min}=0.2， c_2^{max}=1.2 。新产品开发团队选择的综合评价模型相关参数补充如下：$W_q=\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right)^T$， $W_I=\left(\frac{1}{3},\frac{1}{3},\frac{1}{3}\right)^T$；服务质量评价模型中约束信息为：开发时间不超过 500，开发成本不超过 8 000，团队组合的信誉度不低于 0.88。

根据子开发任务的候选团队集合，实验案例总共有 576 种团队组合方式，不同组合方式的综合评价值不同，通过本文模型和算法求解出综合评价值最高的团队组合。以团队组合的知识相似度、协同效应值和服务质量为三维坐标，绘制出案例模型的解空间，如图 7 所示。由于服务质量评价模型中存在相关约束条件，使得团队组合存在非可行解。

根据求解的综合评价结果，优选团队组合为 $TPCS=\left\{CT_1^1,CT_2^3,CT_3^2,CT_4^1,CT_5^2,CT_6^2,CT_7^2,CT_8^1\right\}$，该优选团队组合综合评价值为 $I(TPCS)$=0.785 ，其中知识相似度评价值 $S(TPCS)$=0.821，协同效应评价值 $C(TPCS)$=0.653，服务质量评价值为 $Q(TPCS)$=0.881，各评价值符合实际状况。根据以上评价结果，发现改进人蜂群算法和综合评价模型在解决团队组合优选问题上是可行的。

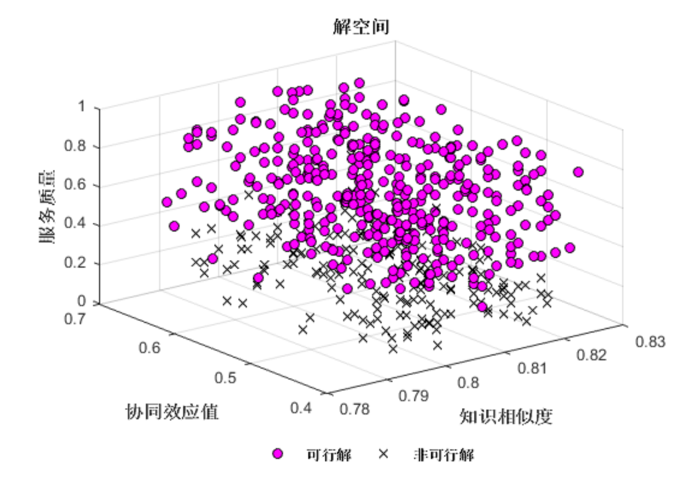


图 7　团队组合解空间

Fig.7　Team combination solution space

表 6　知识交流/返工时间成本矩阵表

Table 6　Knowledge exchange/rework time cost matrix

	CT_1^1	CT_1^2	CT_2^1	CT_2^2	CT_2^3	CT_3^1	CT_3^2	CT_3^3	CT_4^1	CT_4^2	CT_5^1	CT_5^2	CT_6^1	CT_6^2	CT_7^1	CT_7^2	CT_8^1	CT_8^2
CT_1^1	—	—	3/7/12/14	4/8/9/15	3/6/9/13	3/5/13/18	2/3/11/12	3/6/15/27	1/2/12/15	2/2/20/18	1/2/10/16	1/1/8/19	2/3/11/19	1/2/9/16	5/9/13/26	4/8/10/17	7/14/12/20	6/13/16/25
CT_1^2	—	—	4/9/11/18	4/7/12/15	3/7/12/16	4/7/20/13	3/5/18/27	3/6/14/21	2/4/9/19	3/5/22/15	2/3/15/19	1/2/11/22	2/4/18/22	2/3/17/29	6/11/17/20	5/10/16/20	8/15/17/25	7/13/19/23
CT_2^1	3/7/12/14	4/9/11/18	—	—	—	5/10/12/17	6/12/13/14	5/9/14/20	2/4/13/29	3/5/15/22	2/4/13/19	2/5/15/19	2/3/13/17	1/2/14/19	5/10/14/21	4/9/12/17	2/4/8/12	2/5/8/15
CT_2^2	4/8/9/15	4/7/12/15	—	—	—	5/10/17/15	4/9/19/30	6/11/16/18	3/5/12/26	3/6/15/21	2/5/12/17	1/3/10/23	2/3/16/22	2/4/13/25	6/10/18/23	5/9/18/24	2/3/7/15	2/4/10/16

CT_2^3	3/6/9/13	3/7/12/16	—	—	—	6/10/14/20	4/8/10/12	5/9/11/14	2/3/16/17	2/4/17/29	2/3/17/23	1/2/9/14	2/4/11/22	1/1/8/15	5/9/15/16	4/8/9/17	1/3/5/12	1/2/11/17
CT_3^1	3/5/13/18	4/7/20/13	5/10/12/17	5/10/17/15	6/10/14/20	—	—	—	2/3/12/17	2/4/17/15	2/4/10/15	3/5/15/16	3/5/12/16	2/5/8/19	3/6/9/19	4/8/15/22	1/2/11/12	1/2/9/14
CT_3^2	2/3/11/12	3/5/18/27	6/12/13/14	4/9/19/30	4/8/10/12	—	—	—	1/2/8/14	2/3/14/15	2/3/9/15	1/3/6/11	2/4/12/19	2/3/8/14	4/7/13/25	3/5/11/17	2/3/5/10	1/1/6/14
CT_3^3	3/6/15/27	3/6/14/21	5/9/14/20	6/11/16/18	5/9/11/14	—	—	—	2/4/7/19	2/3/9/18	1/3/11/16	2/5/8/14	2/3/13/24	2/4/14/19	4/8/14/22	4/7/10/19	1/2/8/14	2/3/7/15
CT_4^1	1/2/12/15	2/4/9/19	2/4/13/29	3/5/12/26	2/3/16/17	2/3/12/17	1/2/8/14	2/4/7/19	—	—	3/6/9/16	3/5/8/14	3/5/13/18	2/3/15/13	3/5/12/22	2/3/11/20	1/2/8/15	2/4/12/15
CT_4^2	2/2/20/18	3/5/22/15	3/5/15/22	3/6/15/21	2/4/17/29	2/4/17/15	2/3/14/15	2/3/9/18	—	—	4/7/10/21	4/8/11/18	2/4/16/21	3/6/14/19	2/4/12/24	3/6/14/15	1/2/10/19	2/3/13/21
CT_5^1	1/2/10/16	2/3/15/19	2/4/13/19	2/5/12/17	2/3/17/23	2/4/10/15	2/3/9/15	1/3/11/16	3/6/9/16	4/7/10/21	—	—	6/12/15/29	5/10/14/27	2/3/13/25	1/2/14/27	1/1/12/20	1/1/9/21
CT_5^2	1/1/8/19	1/2/11/22	2/5/15/19	1/3/10/23	1/2/9/14	3/5/15/16	1/3/6/11	2/5/8/14	3/5/8/14	4/8/11/18	—	—	5/11/12/24	5/9/11/23	1/2/17/25	1/1/15/19	1/1/10/18	1/2/14/25
CT_6^1	2/3/11/19	2/4/18/22	2/3/13/17	2/3/16/22	2/4/11/22	3/5/12/16	2/4/12/19	2/3/13/24	3/5/13/18	2/4/16/21	6/12/15/29	5/11/12/24	—	—	2/4/10/21	3/5/12/25	1/1/9/18	1/1/9/17
CT_6^2	1/2/9/16	2/3/17/29	1/2/14/19	2/4/13/25	1/1/8/15	2/5/8/19	2/3/8/14	2/4/14/19	2/3/15/13	3/6/14/19	5/10/14/27	5/9/11/23	—	—	3/4/13/24	2/3/12/21	1/1/8/15	1/2/10/21
CT_7^1	5/9/13/26	6/11/17/20	5/10 14/21	6/10/18/23	5/9/15/16	3/6/9/19	4/7/13/25	4/8/14/22	3/5/12/22	2/4/12/24	2/3/13/25	1/2/17/25	2/4/10/21	3/4/13/24	—	—	2/3/7/13	2/4/7/14
CT_7^2	4/8/10/17	5/10/16/20	4/9/12/17	5/9/18/24	4/8/9/17	4/8/15/22	3/5/11/17	4/7/10/19	2/3/11/20	3/6/14/15	1/2/14/27	1/1/15/19	3/5/12/25	2/3/12/21	—	—	1/2/6/12	2/3/8/15
CT_8^1	7/14/12/20	8/15/17/25	2/4/8/12	2/3/7/15	1/3/5/12	1/2/11/12	2/3/5/10	1/2/8/14	1/2/8/15	1/2/10/19	1/1/12/20	1/1/10/18	1/1/9/18	1/1/8/15	2/3/7/13	1/2/6/12	—	—
CT_8^2	6/13/16/25	7/13/19/23	2/5/8/15	2/4/10/16	1/2/11/17	1/2/9/14	1/1/6/14	2/3/7/15	2/4/12/15	2/3/13/21	1/1/9/21	1/2/14/25	1/1/9/17	1/2/10/21	2/4/7/14	2/3/8/15	—	—

4.3　有效性验证和算法对比

在算法构建初期，本文算法直接用综合评价模型的目标函数的值来代替适应度值；随着迭代次数增加，适应度值会逐渐增加到最大适应度值。本文通过对改进前后人工蜂群算法的适应度函数值进行对比分析，进一步分析 MABC 算法在处理团队组合优选问题上的有效性，其实验结果如图 8 所示。

图 8 在一定程度上反映了改进之处对人工蜂群算法的影响，在相同的迭代次数下，MABC 算法具有更高的适应度值；在达到相同的适应度值时，MABC 算法耗时更短。因为改进了人工蜂群算法的搜索方式，在一定程度上加快了算法的收敛速度，避免陷入局部最优解。

为了进一步分析 MABC 算法在处理团队组合优选问题上的可行性和高效性，调用 MATLAB 算法工具箱，引入蚁群算法 (ant colony optimization algorithm,ACO) [18]、遗传算法 (genetic algorithm,GA)。在相同的实验环境下，保持同类型参数设置相同，其中 ACO 算法中信息启发因子、期望启发因子和蒸发系数分别设为 1、2 和 0.8；GA 算法中交叉算子和变异算子分别设置为 0.7 和 0.1。分别进行 100 次实验，得到四种算法处理云环境下新产品开发团队组合优选问题的各项数据，如表 7 所示。通过四种算法的运行结果（均值±标准差）对比，MABC 算法具有求解性能好、求解稳定性好和求解速度快等优势。

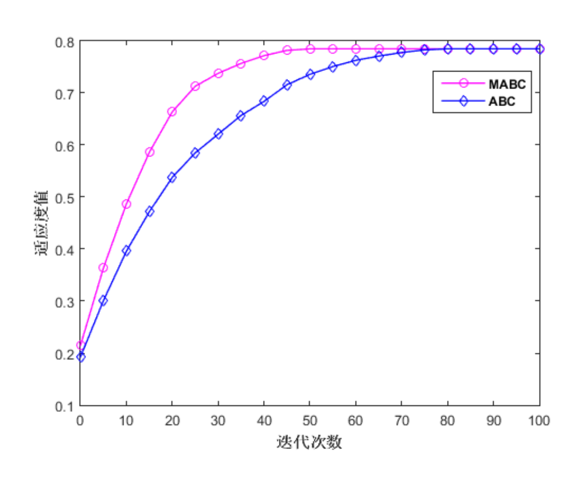


图 8　适应度对比曲线

Fig.8　Fitness curve

表 7 算法运行结果比较

Table 7 Comparison of algorithm operation results

评价指标	算法			
	MABC	ABC	ACO	GA
综合评价值	0.7850±0.0004	0.7841±0.0023	0.7844±0.0019	0.7838±0.0026
知识相似度	0.8210±0.0002	0.8201±0.0028	0.8205±0.0022	0.8199±0.0031
协同效应值	0.6530±0.0003	0.6528±0.0008	0.6529±0.0006	0.6527±0.0011
服务质量评价值	0.8809±0.0011	0.8792±0.0041	0.8798±0.0035	0.8788±0.0045
完成时间(s)	16.1±0.8911	20.3±1.4484	18.2±1.0996	21.9±1.7453

5 结束语

本文通过分析云环境下新产品开发团队组合优选全过程，针对团队组合优选问题，建立了包含知识相似度评价模型、协同效应评价模型和服务质量评价模型的综合评价模型；为了保证算法的求解效果和求解速度，本文引入了人工蜂群算法，并对其适应度函数和搜索方式加以改进。在团队组合优选模型构建过程中，本文既考虑团队自身研发实力，又考虑到团队间的协同效应，也考虑到任务的约束信息。此外，本文通过案例实验与分析，验证了本文的优选方法和模型在解决团队组合优选问题上可行性；通过算法对比，验证了改进算法处理本文模型的有效性。

本文在通过知识相似度来衡量团队与任务之间匹配度，衡量团队研发实力，后续需完善团队实力评价指标，更加精确衡量团队实力；此外，本文采用团队间两两信息交流模式，后续研究中可尝试多团队信息交流，让其更加符合实际。

参考文献：

[1] 李伯虎，张霖，王时龙，等. 云制造—面向服务的网络化制造新模 [J]. 计算机集成制造系统，2010, 16 (1): 1-8. (Li Bohu, Zhang lin, Wang Shilong, *et al.* Cloud manufacturing: a new service-oriented networked manufacturing model [J]. Computer Integrated Manufacturing Systems, 2010, 16 (1): 1-8.)

[2] 刘敬，余隋怀，初建杰，等. 设计云服务平台下网络团队成员优选决策研究 [J]. 计算机集成制造系统，2017, 23 (6): 1205-1215. (Liu Jing, Yu Suihuai, Chu Jianjie, *et al.* Research on member optimal selection of network team [J]. Computer Integrated Manufacturing Systems, 2017, 23 (6): 1205-1215.)

[3] 赵金辉，王学慧. 基于服务质量的云制造服务双向匹配模型 [J]. 计算机集成制造系统，2016, 22 (1): 104-112. (Zhao Jinhui, Wang Xuehui. Two-sided matching model of cloud service based on QoS in cloud manufacturing environment [J]. Computer Integrated Manufacturing Systems, 2016, 22 (1): 104-112.)

[4] 陈友玲，阳玮琦，刘传彪，等. 基于知识相似度和学习能力的产品设计人员指派方法 [J]. 计算机集成制造系统，2017, 23 (8): 1700-1710. (Chen Youling, Yang Weiqi, Liu Chuanbiao, *et al.* Personnel assignment method of product design based on knowledge similarity and learning ability [J]. Computer Integrated Manufacturing Systems, 2017, 23 (8): 1700-1710.)

[5] 姜艳萍，潘恩，梁海明. 考虑指标期望的新产品开发交叉功能团队成员选择方法 [J]. 运筹与管理, 2013, 22 (4): 204-211. (Jiang Yanping, Pan En, Liang Haiming. Method for member selection of cross-functional teams in new product development considering index aspiration [J]. Operations Research And Management Science, 2013, 22 (4): 204-211.)

[6] 王娟茹，赵嵩正. 层次分析法和模糊优先规划法相结合选择虚拟团队成员 [J]. 工业工程, 2007 (6): 105-108. (Wang Juanru, Zhao Songzheng. On selection of partners using AHP and FPP in virtual team [J]. Industrial Engineering Journal, 2007 (6): 105-108.)

[7] Lianying Zhang, Xiang Zhang. Multi-objective team formation optimization for new product development [J]. Computers & Industrial Engineering, 2013, 64: 804-811.

[8] 冯博，樊治平. 基于协同效应的知识创新团队伙伴选择方法 [J]. 管理学报，2012, 9 (2): 258-261. (Feng Bo, Fan Zhiping. A partner selection method for knowledge creation team based on collaborative effect [J]. Chinese Journal of Management, 2012, 9 (2): 258-261.)

[9] Feng Bo, Jiang Zhongzhong, Fan Zhiping, *et al.* A method for member selection of cross-functional teams using the individual and collaborative performances [J]. European Journal of Operational Research, 2010, 203: 652-661.

[10] 徐廷. 基于 Simio 的产品设计过程重叠模式仿真优化研究 [D]. 济南: 山东大学, 2016. (Xu Ting. Research on simulation and optimization of over lapping design process based on Simio [D]. Jinan: Shan dong University, 2016.)

[11] 陈友玲，兰桂花，梁佩馨. 基于知识欧姆定律的上下游活动重叠时间估算 [J]. 计算机集成制造系统, 2016, 22 (12): 2759-2766. (Chen Youling, Lan Guihua, Liang Peixin. Estimation of overlapping time between upstream and downstream activities based on knowledge ohm’s law [J]. Computer Integrated Manufacturing Systems, 2016, 22 (12): 2759-2766.)

[12] 马文建，刘伟，李传昭. 并行产品开发中设计活动间重叠与信息交流 [J]. 计算机集成制造系统, 2008 (4): 630-636. (Ma Wenjian, Liu Wei, Li Chuanzhao. Overlapping and communication frequency between activities within concurrent product development [J]. Computer Integrated Manufacturing Systems, 2008 (4): 630-636.)

[13] 吕炎杰，赵罡，于勇. 基于模糊集理论和知识相似度的复杂产品设计任务分配方法 [J]. 计算机集成制造系统，2015, 21 (4): 904-913. (Lyu Yanjie, Zhao Gang, Yu Yong. Task assignment method of complex product design based on fuzzy set theory and knowledge similarity [J]. Computer Integrated Manufacturing Systems, 2015, 21 (4): 904-913.)



[14] 李靖, 张永安. 基于 ISM 的物流网络协同效应影响因素分析——以苏宁电器为核心的物流网络为例 [J]. 北京交通大学学报: 社会科学版, 2011 (4): 45-52. (Li Jing, Zhang Yongan. An analysis of the affecting factors of collaborative effect of the logistics networks: a case study of Suning’s core logistics networks [J]. Journal of Beijing Jiaotong University: Social Sciences Edition, 2011 (4): 45-52.)

[15] 任子武, 王振华, 孙立宁. 基于改进人工蜂群算法的并联机器人正运动学解 [J]. 机械工程学报, 2013, 49 (13): 48-55. (Ren Ziwu, Wang Zhenhua, Sun Lining. Forward kinematics solution for parallel manipulators based on improved artificial bee colony algorithm [J]. Journal of Mechanical Engineering, 2013, 49 (13): 48-55.)

[16] Liang Yaosheng, Wan Zhongping, Fang Debin. An improved artificial bee colony algorithm for solving constrained optimization problems [J]. International Journal of Machine Learning and Cybernetics, 2017, 3 (8): 739-754.

[17] 李彦苍, 彭扬. 基于信息熵的改进人工蜂群算法 [J]. 控制与决策, 2015, 30 (6): 1121-1125. (Li Yancang, Peng Yang. Improved artificial bee colony algorithm based on information entropy [J]. Control and Decision, 2015, 30 (6): 1121-1125.)

[18] 夏亚梅, 程渤, 陈俊亮, 等. 基于改进蚁群算法的服务组合优化 [J]. 计算机学报, 2012, 35 (2): 2270-2281. (Xia Yamei, Chen Bo, Chen Junliang, *et al*. Optimizing services composition based on improved ant colony algorithm [J]. Chinese Journal of Computers, 2012, 35 (2): 2270-2281.)